Column	Label	Formula	Explanation
			Segment B Total Structure $ multiplied by the Fiber Structure %.
EA	Aerial Fiber Main Feeder Structure $	=IF(CH2="YES",0,CV2*BG2)	If the Main Feeder segment is Copper only Aerial Fiber Main Feeder Structure $ equals zero. If not copper only then the Aerial Fiber Main Feeder Structure $ equals the Main Feeder Aerial Segment B Total Structure $ multiplied by the Fiber Structure %.
EB	Total Fiber Main Feeder Structure $	=SUM(DY2:EA2)	Total Fiber Main Feeder Structure $ equals the sum of Underground, Buried and Aerial Main Feeder Structure $.
EC	Main Feeder Fiber Cost per Fiber	=IF(CB2=0,0,(DX2+EB2)/CB2)	Main Feeder Fiber Cost per Fiber equals zero if the Number of Main Feeder Segment Fibers Required equals zero. If the Number of Main Feeder Segment Fibers Required is not zero then Main Feeder Fiber Cost equals the sum of Total Fiber Main Feeder Material $ and Total Fiber Main Feeder Structure $ divided by the Number of Main Feeder Segment Fibers Required.
ED	Main Feeder Fiber Cumulative Cost per Fiber	=SUM(INDIRECT("ec"&ROW(EC2)&":ec"&MATCH($AN2,CliiQuad,0)+1))	This calculation accumulates the cost per Fiber of the CBGs in a Quadrant from the office to the current CBG.
EE	Main Feeder Fiber Costs Allocated to this CBG for DLC-S	=BX2*ED2	Main Feeder Fiber Costs Allocated to this CBG for DLC-S equals the # DLC-S fibers this BG multiplied by Main Feeder Fiber Cumulative Cost per Fiber.
EF	Main Feeder Fiber Costs Allocated to this CBG for DLC-L	=BZ2*ED2	Main Feeder Fiber Costs Allocated to this CBG for DLC-L equals the # DLC-L fibers this BG multiplied by Main Feeder Fiber Cumulative Cost per Fiber.
EG	Underground Fiber Subfeeder Distance	=IF(CE2="N/A",0,VLOOKUP(U2,FiberPlantMixTable,2)*AA2)	If Subfeeder Fiber Size equals "N/A" then Underground Fiber Subfeeder Distance equals zero. If not zero then Underground Fiber Subfeeder Distance equals the "A" Sub Feeder Distance multiplied by the density specific underground percent of fiber.
EH	Underground Fiber Subfeeder Material $	=IF(EG2=0,0,EG2*(VLOOKUP(CE2,FiberCableCost,2,FALSE)*FiberCostRatio))	If Underground Fiber Subfeeder Distance is zero then Underground Fiber Subfeeder Material $ equals zero. If not zero the Underground Fiber Subfeeder Material $ is calculated by

Column	Label	Formula	Explanation
			multiplying the Underground Fiber Subfeeder Distance by the cost for the appropriate fiber cable size plus the cost associated with the Number of Main Feeder Maximum Size Fiber Cables. The total is then multiplied by Fiber Cost Ratio or Discount (default value = 0).
EI	Buried Fiber Subfeeder Distance	=IF(CE2="N/A",0,VLOOKUP(U2,FiberPlantMixTable,3)*AA2)	If Subfeeder Fiber Size equals "N/A" then Buried Fiber Subfeeder Distance equals zero. If not zero then Buried Fiber Subfeeder Distance equals the "A" Sub Feeder Distance multiplied by the density specific Buried percent of fiber.
EJ	Buried Fiber Subfeeder Material $	=IF(EI2=0,0,EI2*(VLOOKUP(CE2,FiberCableCost,3,FALSE)*FiberCostRatio))	If Buried Fiber Subfeeder Distance is zero then Buried Fiber Subfeeder Material $ equals zero. If not zero the Buried Fiber Subfeeder Material $ is calculated by multiplying the Buried Fiber Subfeeder Distance by the cost for the appropriate fiber cable size plus the cost associated with the Number of Main Feeder Maximum Size Fiber Cables. The total is then multiplied by Fiber Cost Ratio or Discount (default value = 0).
EK	Aerial Fiber Subfeeder Distance	=IF(CE2="N/A",0,VLOOKUP(U2,FiberPlantMixTable,4)*AA2)	If Subfeeder Fiber Size equals "N/A" then Aerial Fiber Subfeeder Distance equals zero. If not zero then Aerial Fiber Subfeeder Distance equals the "A" Sub Feeder Distance multiplied by the density specific Aerial percent of fiber.
EL	Aerial Fiber Subfeeder Material $	=IF(EK2=0,0,EK2*(VLOOKUP(CE2,FiberCableCost,4,FALSE)*FiberCostRatio))	If Aerial Fiber Subfeeder Distance is zero then Aerial Fiber Subfeeder Material $ equals zero. If not zero the Aerial Fiber Subfeeder Material $ is calculated by multiplying the Aerial Fiber Subfeeder Distance by the cost for the appropriate fiber cable size plus the cost associated with the Number of Main Feeder Maximum Size Fiber Cables. The total is then multiplied by Fiber Cost Ratio or Discount (default value = 0).
EM	Total Fiber Subfeeder Material $	=EH2+EJ2+EL2	Total Fiber Subfeeder Material $ equals the sum of Underground Fiber Subfeeder Material $, Buried Fiber Subfeeder Material $ and Aerial Fiber Subfeeder Material $.
EN	Underground Fiber Subfeeder Structure $	=IF(CE2="N/A",0,DL2)	If Subfeeder Fiber Size equals "N/A" then Underground Fiber Structure $ equals zero. If not zero, then Underground Fiber Subfeeder Structure $ equals Total Subfeeder Segment A Underground Structure $.

Column	Label	Formula	Explanation
EV	DLC-S on Fiber Electronics $	=IF(AW2="DLC-S",DLC_SDiscount*BT2*(VLOOKUP(BO2,DigitalCarrierCost,2)+BO2*VLOOKUP(BO2,DigitalCarrierCost,3)),0)	If Segment Type 1 equals DLC-S then DLC-S on Fiber Electronics $ equals (1-% DLC-S Discount) * the (Total # Terminal locations in CBG * the fixed cost for the # Voice Grade Lines Equipped per Terminal Location) + the total per line cost for the # Voice Grade Lines per Terminal Location. If Segment type 1 does not equal DLC-S then DLC-S on Fiber Electronics $ equals zero.
EW	Total DLC-S on Fiber $	=EE2+EU2+EV2	Total DLC-S on Fiber $ equals the sum of Main Feeder Fiber Cost Allocated to this CBG for DLC-S, Total Subfeeder Fiber $ assigned DLC-S and DLC-S on Fiber Electronics $.
EX	Number of FDI's per location at maximum capacity	=IF(BO2>3600,INT(BO2/3600),0)	If the # of Voice Grade Lines Equipped per Terminal Location is greater than 3600 Then the Number of FDI's per location at maximum capacity equals the integer value of the # of Voice Grade Lines Equipped per Terminal Location divided by 3600. If the # of Voice Grade Lines Equipped is less than 3600 then the Number of FDI's per location at maximum capacity is zero.
EY	Feeder Distribution Interface Cost	=BT2*(EX2*CapacityFDIcost+IF(BO2=0,0,VLOOKUP(BO2-(3600*EX2),FdiCostTable,2)))	Feeder Distribution Interface Cost equals the Total # Terminal Location in CBG * (the Number of FDI's per Location at Capacity * the total per line FDI capacity cost + the residual # Voice Grade Lines Equipped per Terminal Location multiplied by the per line FDI cost).
EZ	Subtotal Feeder Cost	=IF(V2=0,0,DQ2+ET2+EW2+EY2)	If Total CBG Lines Served equals zero then Subtotal Feeder Cost equals zero. If Total CBG lines Served is not zero then Subtotal feeder cost is the sum of Total Copper & T1 Related Feeder Cost Assigned this CBG, Total DLC-L on Fiber $, Total DLC-S on Fiber $ and Feeder Distribution Interface Cost.
FA	Fiber Feeder Extension Part 1 Length	=((AL2-1)/AL2)*Y2	This formula calculates the length of the vertical extension of the feeder in the CBG. It equals ((the Number of Feeder legs in CBG - 1)/Number Feeder Legs in CBG) * Segment D.
FB	Fiber Feeder Extension Part 1 # Fibers	=(INT(AL2/2))*BV2	The Fiber Feeder Extension Part 1 # Fibers equals the integer value of the Number of Feeder Legs in CBG/2 multiplied by the Total Number of Fibers per Feeder Leg.

Column	Label	Formula	Explanation
FC	# Max Size Fiber Feeder Extension Part 1 Cables	=TRUNC(IF(FB2>MaxFiberSize,FB2/MaxFiberSize,0))	If the Fiber Feeder Extension Part 1 # Fibers is greater than the maximum fiber cable size then the # Max Size Fiber Feeder Extension Part 1 Cables equals the truncated value of Fiber Feeder Extension Part 1 # fibers divided by the max. fiber cable size. The formula returns a zero if the Fiber Feeder Extension Part 1 # Fibers is less than the max. fiber cable size.
FD	Last Fiber Fiber Feeder Extension Part 1 Cable Size	=IF(FB2>0,INDEX(FiberCableSize,MATCH(FB2-(MaxFiberSize*FC2),FiberCableSize,-1),1),0)	This formula calculates the size of the cable required for the residual number of Fiber Feeder Extension Part 1 # Fibers if the Fiber Feeder Extension Part 1 # Fibers is greater than zero.
FE	Aerial Fiber Feeder Extension Part 1 Distance	=VLOOKUP(U2,FiberPlantMixTable,4)*FA2	This formula calculates the distance of the aerial portion of the vertical extension of feeder in the CBG. The Fiber Feeder Extension Part 1 Length is multiplied by the density specific percent of aerial cable.
FF	Aerial Fiber Feeder Extension Part 1 Material $	=IF(FE2=0,0,FE2*(VLOOKUP(FD2,FiberCableCost,4,FALSE)+(FC2*VLOOKUP(MaxFiberSize,FiberCableCost,4))))*FiberCostRatio	This formula calculates the Aerial fiber cable cost less discount, if applicable, of the vertical extension of the feeder in the CBG if the Aerial Fiber Feeder Extension Part 1 Distance does not equal zero.
FG	Buried Fiber Feeder Extension Part 1 Distance	=VLOOKUP(U2,FiberPlantMixTable,3)*FA2	This formula calculates the distance of the buried portion of the vertical extension of feeder in the CBG. The Fiber Feeder Extension Part 1 Length is multiplied by the density specific percent of buried cable.
FH	Buried Fiber Feeder Extension Part 1 Material $	=IF(FG2=0,0,FG2*(VLOOKUP(FD2,FiberCableCost,3,FALSE)+(FC2*VLOOKUP(MaxFiberSize,FiberCableCost,3))))*FiberCostRatio	This formula calculates the Buried fiber cable cost less discount, if applicable, of the vertical extension of the feeder in the CBG if the Buried Fiber Feeder Extension Part 1 Distance does not equal zero.
FI	Underground Fiber Feeder Extension Part 1 Distance	=VLOOKUP(U2,FiberPlantMixTable,2)*FA2	This formula calculates the distance of the underground portion of the vertical extension of feeder in the CBG. The Fiber Feeder Extension Part 1 Length is multiplied by the density specific percent of underground cable.
FJ	Underground Fiber Feeder Extension Part 1 Material $	=IF(FI2=0,0,FI2*(VLOOKUP(FD2,FiberCableCost,2,FALSE)+(FC2*VLOOKUP(MaxFiberSize,FiberCableCost,2))))*FiberCostRatio	This formula calculates the Underground fiber cable cost less discount, if applicable, of the vertical extension of the feeder in the CBG if the Underground Fiber Feeder Extension Part 1 Distance does not equal zero.

Column	Label	Formula	Explanation
FK	Total Feeder Extension Part 1 Material $	=FF2+FH2+FJ2	Total Feeder Extension Part 1 Material $ equals the sum of Aerial Fiber Feeder Extension Part 1 Material $, Buried Fiber Feeder Extension Part 1 Material $ and Underground Fiber feeder Extension Part 1 Material $.
FL	Aerial Feeder Extension Part 1 Structure $	=IF(FE2<1,0,ROUND(FE2/VLOOKUP(U2,SpacingTable,3),1)+IF(VLOOKUP(U2,CopperPlantMixTable,4)<0.005,0,1))*AI2*(IF(AH2=1,VLOOKUP(U2,HardRockStructure,6),IF(OR(AH2=2,AF2=1),VLOOKUP(U2,SoftRockStructure,6),VLOOKUP(U2,NormalStructure,6))))	This formula calculates density specific Aerial Feeder structure cost for the vertical feeder extension in the CBG if the Aerial Fiber Feeder Extension Part 1 Distance does not equal zero. The formula returns a value of zero if the distance equals zero. Structure costs include density specific terrain costs.
FM	Buried Feeder Extension Part 1 Structure $	=IF(FG2<1,0,AI2*FG2*IF(AH2=1,VLOOKUP(U2,HardRockStructure,4),IF(OR(AH2=2,AF2=1),VLOOKUP(U2,SoftRockStructure,4),VLOOKUP(U2,NormalStructure,4))))	This formula calculates density specific Buried Feeder structure cost for the vertical feeder extension in the CBG if the Buried Fiber Feeder Extension Part 1 Distance does not equal zero. The formula returns a value of zero if the distance equals zero. Structure costs include density specific terrain costs.
FN	Underground Feeder Extension Part 1 Structure $	=IF(FI2<1,0,AI2*((ROUND(FI2/VLOOKUP(U2,SpacingTable,2),1)+1)*VLOOKUP(0,ConduitManholeTable,VLOOKUP(AH2,SurfaceConditionTable,2))+FI2*(2*Conduitperductfoot+IF(AH2=1,VLOOKUP(U2,HardRockStructure,2),IF(OR(AH2=2,AF2=1),VLOOKUP(U2,SoftRockStructure,2),VLOOKUP(U2,NormalStructure,2))))))	This formula calculates density specific Underground Feeder structure cost for the vertical feeder extension in the CBG if the Underground Fiber Feeder Extension Part 1 Distance does not equal zero. The formula returns a value of zero if the distance equals zero. Structure costs include density specific terrain costs
FO	Total Feeder Extension Part 1 Structure $	=SUM(FL2:FN2)	Total Feeder Extension Part 1 Structure $ equal the sum of Aerial, Buried and Underground Feeder Extension Part 1 Structure $.
FP	Total Cost Feeder Extension Part 1	=FK2+FO2	Total Cost Feeder Extension Part 1 equals Total Feeder Extension Part 1 Material $ plus Total Feeder Extension Part 1 Structure $.
FQ	# Max Size Fiber Feeder Extension Part 2 Cables	=TRUNC(IF(BV2>MaxFiberSize,BV2/MaxFiberSize,0))	If the Fiber Feeder Extension Part 2 # Fibers is greater than the maximum fiber cable size then the # Max Size Fiber Feeder Extension Part 2 Cables equals the truncated value of Fiber Feeder Extension Part 2 # fibers divided by the max. fiber cable size. The formula returns a zero if the Fiber Feeder Extension Part 2 # Fibers is less than the max. fiber cable size.

Column	Label	Formula	Explanation
FR	Last Fiber Feeder Extension Part 2 Cable Size	=IF(BV2>0,INDEX(FiberCableSize,MATCH((BV2-(MaxFiberSize*FQ2)),FiberCableSize,-1),1),0)	This formula calculates the size of the residual Fiber Feeder Extension Part 2, which is the horizontal leg of the feeder in the CBG, if the Total Number of Fibers per feeder Leg is greater than zero. It determines the residual number of fibers subtracting the number of fibers in the maximum size cables from the Total Number of Fibers per Feeder Leg. It then selects the smallest cable size greater than the number of residual fibers.
FS	Size Feeder Extension Part 2 (If Copper)	=IF(AW2="Cable",INDEX(FeederCableSize,MATCH(((V2-AV2+AV2/12)/AL2)/VLOOKUP(U2,DensityFillTable,2),FeederCableSize,-1),1),0)	If Segment Type 1 is cable this formula determines the size of the Feeder Extension by taking Total CBG Lines served and converting those Lines in the CBG Provisioned as DS-1s to Copper T-1 equivalents. It then divides the number of the adjusted lines served by the Number of Feeder legs in CBG to determine the number of lines per feeder leg. The number of lines per feeder leg is then divided by the a density specific fill factor to determine the required cable size. The selects the smallest cable size greater than the number of lines calculated. If Segment Type 1 does not equal cable the formula returns a zero value.
FT	Aerial Feeder Extension Part 2 Distance	=IF(AW2="Cable",VLOOKUP(U2,CopperPlantMixTable,4),VLOOKUP(U2,FiberPlantMixTable,4))*BM2	If Segment Type 1 equals Cable then this formula multiplies the Horizontal Fiber Feeder Cable Length by the percent of aerial copper cable to determine the distance. If Segment Type 1 does not equal zero it multiplies the Horizontal Fiber Feeder Cable Length by the percent of Aerial fiber cable to determine distance.
FU	Aerial Copper Feeder Extension Part 2 Material $	=IF(FS2=0,0,FT2*AI2*AL2*CopperCostRatio*VLOOKUP(FS2,FeederCableCost,4,FALSE))	This formula calculates the Aerial copper cable cost less discount, if applicable, including adjustment by the New Terrain and Water Cost Multiplier of the horizontal extension of the feeder in the CBG if the Size Feeder Extension Part 2(If Copper) does not equal zero. The Formula returns a zero value if Size Feeder Extension Part 2 (If Copper) equals zero.
FV	Aerial Fiber Feeder Extension Part 2 Material $	=IF(OR(FT2=0,FR2=0),0,FT2*(VLOOKUP(FR2,FiberCableCost,4,FALSE)+(FQ2*VLOOKUP(MaxFiberSize,FiberCableCost,4))))*AL2*FiberCostRatio	This formula calculates the Aerial Fiber cable cost less discount, if applicable, of the horizontal extension of the feeder in the CBG if the Aerial Feeder Extension Part 2 Distance does not equal zero.
FW	Buried Feeder Extension Part 2 Distance	=IF(AW2="Cable",VLOOKUP(U2,CopperPlantMixTable,3),VLOOKUP(U2,FiberPlantMixTable,3))*BM2	If Segment Type 1 equals Cable then this formula multiplies the Horizontal Fiber Feeder Cable Length by the density specific percent of buried copper cable to determine the distance. If

Column	Label	Formula	Explanation
			Segment Type 1 does not equal cable it multiplies the Horizontal Fiber Feeder Cable Length by the density specific percent of buried fiber cable to determine distance.
FX	Buried Copper Feeder Extension Part 2 Material $	=IF(FS2=0,0,FW2*AI2*AL2*CopperCostRatio*VLOOKUP(FS2,FeederCableCost,3,FALSE))	This formula calculates the Buried copper cable cost less discount, if applicable, of the horizontal extension of the feeder in the CBG if the Size Feeder Extension Part 2 (If Copper) does not equal zero.
FY	Buried Fiber Feeder Extension Part 2 Material $	=IF(OR(FW2=0,FR2=0),0,FW2*(VLOOKUP(FR2,FiberCableCost,3,FALSE)+(FQ2*VLOOKUP(MaxFiberSize,FiberCableCost,3))))*AL2*FiberCostRatio	This formula calculates the Buried Fiber cable cost less discount, if applicable, of the horizontal extension of the feeder in the CBG if the Buried Feeder Extension Part 2 Distance does not equal zero.
FZ	Underground Feeder Extension Part 2 Distance	=IF(AW2="Cable",VLOOKUP(U2,CopperPlantMixTable,2),VLOOKUP(U2,FiberPlantMixTable,2))*BM2	If Segment Type 1 equals Cable then this formula multiplies the Horizontal Fiber Feeder Cable Length by the density specific percent of underground copper cable to determine the distance. If Segment Type 1 does not equal cable it multiplies the Horizontal Fiber Feeder Cable Length by the density specific percent of underground fiber cable to determine distance.
GA	Underground Copper Feeder Extension Part 2 Material $	=IF(FS2=0,0,FZ2*AI2*AL2*CopperCostRatio*VLOOKUP(FS2,FeederCableCost,2,FALSE))	This formula calculates the underground copper cable cost less discount, if applicable, of the horizontal extension of the feeder in the CBG if the Size Feeder Extension Part 2 (If Copper) does not equal zero. The formula returns a zero value if Size Feeder Extension Part 2 (If Copper) equals zero.
GB	Underground Fiber Feeder Extension Part 2 Material $	=IF(OR(FZ2=0,FR2=0),0,FZ2*(VLOOKUP(FR2,FiberCableCost,2,FALSE)+(FQ2*VLOOKUP(MaxFiberSize,FiberCableCost,2))))*AL2*FiberCostRatio	This formula calculates the Underground Fiber cable cost less discount, if applicable, of the horizontal extension of the feeder in the CBG if the Underground Feeder Extension Part 2 Distance does not equal zero. The formula returns a zero value if Underground Feeder Extension Part 2 Distance equals zero.
GC	Total Copper Feeder Extension Part 2 Material $	=FU2+FX2+GA2	Total Copper Feeder Extension Part 2 Material $ equals the sum of Aerial Copper Feeder Extension Part 2 Material $, Buried Copper Feeder Extension Part 2 Material $ and Underground Copper Feeder Extension Part 2 Material $.
GD	Total Fiber Feeder Extension Part 2 Material $	=FV2+FY2+GB2	Total Fiber Feeder Extension Part 2 Material $ equals the sum of Aerial Fiber Feeder Extension Part 2 Material $, Buried Fiber Feeder Extension Part 2 Material $ and Underground Fiber Feeder

Column	Label	Formula	Explanation
			Extension Part 2 Material $.
GE	Aerial Feeder Extension Part 2 Structure $	=IF(FT2<1,0,ROUND((FT2*AL2)/VLOOKUP(U2,SpacingTable,3),1)+IF(VLOOKUP(U2,CopperPlantMix Table,4)<0.005,0,1))*AI2*(IF(AH2=1,VLOOKUP(U2,HardRockStructure,6),IF(OR(AH2=2,AF2=1),VLOOKUP(U2,SoftRockStructure,6),VLOOKUP(U2,NormalStructure,6))))	This formula calculates density specific Aerial Feeder structure cost for the horizontal feeder extension in the CBG if the Aerial Feeder Extension Part 2 Distance does not equal zero. The formula returns a value of zero if the distance equals zero. Structure costs include density specific terrain costs
GF	Buried Feeder Extension Part 2 Structure $	=IF(FW2<1,0,AI2*FW2*AL2*IF(AH2=1,VLOOKUP(U2,HardRockStructure,4),IF(OR(AH2=2,AF2=1),VLOOKUP(U2,SoftRockStructure,4),VLOOKUP(U2,NormalStructure,4))))	This formula calculates density specific Buried Feeder structure cost for the horizontal feeder extension in the CBG if the Buried Feeder Extension Part 2 Distance does not equal zero. The formula returns a value of zero if the distance equals zero. Structure costs include density specific terrain costs
GG	Underground Feeder Extension Part 2 Structure $	=IF(FZ2<1,0,AI2*FZ2*AL2*IF(AH2=1,VLOOKUP(U2,HardRockStructure,2),IF(OR(AH2=2,AF2=1),VLOOKUP(U2,SoftRockStructure,2),VLOOKUP(U2,NormalStructure,2))))	This formula calculates density specific Underground Feeder structure cost for the horizontal feeder extension in the CBG if the Underground Feeder Extension Part 2 Distance does not equal zero. The formula returns a value of zero if the distance equals zero. Structure costs include density specific terrain costs.
GH	Total Feeder Extension Part 2 Structure $	=SUM(GE2:GG2)	Total Feeder Extension Part 2 Structure $ are the sum of Aerial Feeder Extension Part 2 Structure $, Buried Feeder Extension Part 2 Structure $ and Underground Feeder Extension Part 2 Structure $.
GI	Total Feeder Extension Part 2 $	=GC2+GD2+GH2	Total Feeder Extension Part 2 $ is the sum of Total Copper Feeder Extension Part 2 Material $, Total Fiber Feeder Extension Material $ and Total Feeder Extension Part 2 Structure $.
GJ	Grand Total Feeder Cost	=EZ2+FP2+GI2	Grand Total Feeder Cost is the sum of Subtotal feeder cost, Total Cost Feeder Extension Part 1 and Total Feeder Extension Part 2 $.
GK	# Pairs required per Distribution Leg	=CEILING((((V2-AV2+AV2/12)/AL2)/VLOOKUP(U2,DensityFillTable,3))/(AR2*2),1)	The # Pairs required per Distribution Leg equals the ((((Total CBG Lines Served minus the lines in CBG Provisioned as DS-1s plus Lines in CBG Provisioned as DS-1s/12)/Number of Feeder Legs in CBG)/Density specific Fill)/(Number of Distribution Legs *2).

Column	Label	Formula	Explanation
GL	# Max size Distribution Leg Cables	=IF(GK2>0,TRUNC(IF(GK2>MaxDistrSize,GK2/MaxDistrSize,0)),0)	If the # Pairs required per Distribution Leg is greater than zero then the # Max size Distribution Leg Cables equals the truncated # Pairs required per Distribution Leg divided by the maximum size distribution cable (3600) if the # of Pairs per Distribution Leg is greater than 3600. The formula returns a zero value if neither of the conditional test are true.
GM	Residual Copper Distribution Leg Cable Size	=IF(GK2>0,INDEX(DistrCableSize,MATCH(GK2-(MaxDistrSize*GL2),DistrCableSize,-1),1),0)	This formula determines the cable size required for the residual copper Distribution pairs, i.e. those pairs in excess of those served by the maximum distribution size cables.
GN	Fiber Distribution Leg Cable size	=IF(AND(NOT(AW2="Cable"),GM2=0),INDEX(FiberCableSize,MATCH(BU2/(2*AS2),FiberCableSize,-1),1),0)	If Segment Type 1 does not equal Cable and Copper Distribution Leg Cable Size equals zero then this formula determines the Fiber Distribution Leg Cable size. The cable size selected will be that cable which is the smallest cable greater than (# fibers Required per Terminal Location/(2*Number of Distribution Vertical Legs per terminal). If Segment type 1 equals Cable and Copper Distribution Leg Cable Size is not zero the formula returns a zero value.
GO	# Pairs per Horizontal Distribution Leg	=CEILING((((V2-AV2+AV2/12)/AL2)/VLOOKUP(U2,DensityFillTable,3))/(AT2*2),1)	The # Pairs per Horizontal Distribution Leg equals the ((((Total CBG Lines Served minus the lines in CBG Provisioned as DS-1s plus Lines in CBG Provisioned as DS-1s/12)/Number of Feeder Legs in CBG)/Density specific Fill)/(Number of Terminal Locations per Feeder Leg*2). The result of this formula is rounded up.
GP	Last Horizontal Distribution Copper Cable Size	=IF(GM2>0,INDEX(DistrCableSize,MATCH(GO2-(MaxDistrSize*GQ2),DistrCableSize,-1),1),0)	This formula calculates the residual horizontal cable size if Copper Distribution Leg Cable Size is greater than zero. The cable size is the smallest size great than # pairs per Horizontal Distribution Leg minus the maximum distribution cable size multiplied # Max Horizontal Copper Distribution Cables. Last Horizontal Distribution Leg equals zero if Copper Distribution Leg Cable Size equals zero.
GQ	# Max Horizontal Copper Distribution Cables	=IF(GM2>0,TRUNC(IF(GO2>MaxDistrSize,GO2/MaxDistrSize,0)),0)	If the Copper Distribution Leg Cable Size is greater than zero and if it is greater than Maximum Distribution cable Size then the # Max Horizontal Copper Distribution Cables equals # Pairs per Horizontal Distribution Leg divided by the Maximum Distribution Cable Size.

Column	Label	Formula	Explanation
GR	Last Horizontal Fiber Distribution Cable Size	=IF(GN2>0,INDEX(FiberCableSize,MATCH((BU2/2)-(MaxFiberSize*GS2),FiberCableSize,-1),1),0)	This formula calculates the residual Horizontal Fiber Distribution Cable Size if the Fiber Distribution Leg Cable Size is greater than zero. The cable size equals the # Fibers Required per Terminal Location divided by 2 minus # Max Size Horizontal Fiber Distribution Cables multiplied by the maximum fiber cable size.
GS	# Max Size Horizontal Fiber Distribution Cables	=IF(GN2>0,TRUNC(IF((BU2/2)>MaxFiberSize,(BU2/2)/MaxFiberSize,0)),0)	If the Fiber Distribution Leg Cable Size is greater than zero then the # Max Size Horizontal Fiber Distribution Cables equals the (# Fibers Required per Terminal Location/2)/maximum fiber cable size if (# Fibers Required per Terminal Location /2) is greater than MaxFiber Size. If the Fiber Distribution Leg Cable Size equals zero or (# Fibers Required per Terminal Location/2) is less than MaxFiberSize the formula returns a zero value.
GT	Feeder Length within CBG	=BM2+FA2/2-(AD2*AQ2*(AT2-1)/2)	Feeder Length within CBG equals the Horizontal Fiber Feeder Cable Length plus Fiber Feeder Extension Part 1 Length divided by 2.
GU	Total Feeder Length	=W2+AA2+GT2	Total Feeder Length is the sum of Total "B" Distance, "A" Sub Feeder Distance and Feeder Length within CBG.
GV	Total Distribution Length	=BL2+AO2/2	Total Distribution Length equals Longest Actual Horizontal Copper Distribution Distance plus Distribution Vertical Distance per Feeder divided by 2.
GW	Total # Pedestals in CBG	=IF(J2="",0,AR2*AR2)	Total # Pedestals in CBG equals the Number of Distribution Legs squared.
GX	Length To Center of a Lot	=AD2*0.5*SQRT(2)	Length to the Center of a Lot equals Base Lot Side Length multiplied by .5 multiplied by the square root of 2.
GY	Total Aerial Distribution Horizontal Distance	=IF(BN2=0,0,BN2*VLOOKUP(U2,DistributionPlantMixTable,4))	Total Aerial Distribution Horizontal Distance equals Total Horizontal Copper Cable Length multiplied by the density specific percentage of aerial cable. If Total Horizontal cable length is zero then Total Aerial Distribution Horizontal Distance is zero.
GZ	Total Aerial Distribution Vertical Distance	=AL2*AO2*AR2*VLOOKUP(U2,DistributionPlantMixTable,4)	Total Aerial Distribution Vertical Distance equals the Number of Feeder Legs in CBG multiplied by the Distribution Vertical

Column	Label	Formula	Explanation
			Distance per Feeder multiplied by the Number of Distribution legs multiplied by the density specific percentage of Aerial cable.
HA	Aerial Distribution Copper Material $	=CopperCostRatio*(IF(GM2=0,0,GZ2*(VLOOKUP(GM2,DistrCableCost,4,FALSE)+GL2*VLOOKUP(MaxDistrSize,DistrCableCost,4)))+IF(GP2=0,0,GY2*(VLOOKUP(GP2,DistrCableCost,4,FALSE)+GQ2*VLOOKUP(MaxDistrSize,DistrCableCost,4))))	Aerial Distribution Copper material $ equals Total Aerial Distribution Vertical Distance multiplied by the Distribution cable cost for Copper Distribution Leg Cable Size plus Total Aerial Distribution Horizontal Distance multiplied by the LastHorizontal Distribution Copper Cable Size. The sum of these cable costs are multiplied by 1 minus the copper cable cost discount, if applicable. The formula returns a zero value for the vertical cable cost if the Copper Distribution Leg Cable Size equals zero and for the Horizontal cable cost if the Last Horizontal Distribution Cable Size equals zero.
HB	Total Buried Distribution Horizontal Distance	=IF(BN2=0,0,BN2*VLOOKUP(U2,DistributionPlantMixTable,3))	Total Buried Distribution Horizontal Distance is calculated by multiplying the Total Horizontal Copper Cable Length by the density specific percentage of buried cable.
HC	Total Buried Distribution Vertical Distance	=AL2*AO2*AR2*VLOOKUP(U2,DistributionPlantMixTable,3)	Total Buried Distribution Vertical Distance is calculated by multiplying the Number of Feeder Legs in CBG by Distribution Vertical Distance per Feeder by the Number of Distribution legs by the density specific percentage of aerial cable.
HD	Buried Distribution Copper Material $	=CopperCostRatio*(IF(GM2=0,0,HC2*VLOOKUP(GM2,DistrCableCost,3,FALSE)+GL2*VLOOKUP(MaxDistrSize,DistrCableCost,3))+IF(GP2=0,0,HB2*VLOOKUP(GP2,DistrCableCost,3,FALSE)+GQ2*VLOOKUP(MaxDistrSize,DistrCableCost,3)))	Buried Distribution Copper material $ equals Total Buried Distribution Vertical Distance multiplied by the Distribution cable cost for Copper Distribution Leg Cable Size plus Total Buried Distribution Horizontal Distance multiplied by the LastHorizontal Distribution Copper Cable Size. The sum of these cable costs are multiplied by 1 minus the copper cable cost discount, if applicable. The formula returns a zero value for the vertical cable cost if the Copper Distribution Leg Cable Size equals zero and for the Horizontal cable cost if the Last Horizontal Distribution Cable Size equals zero
HE	Total Underground Distribution Horizontal Distance	=IF(BN2=0,0,BN2*VLOOKUP(U2,DistributionPlantMixTable,2))	Total Underground Distribution Horizontal Distance equals Total Horizontal Copper Cable Length multiplied by the density specific percentage of underground cable. If Total Horizontal cable length is zero then Total Underground Distribution Horizontal Distance is zero.

Column	Label	Formula	Explanation
HF	Total Underground Distribution Vertical Distance	=AL2*AO2*AR2*VLOOKUP(U2,DistributionPlantMix Table,2)	Total Underground Distribution Vertical Distance equals the Number of Feeder Legs in CBG multiplied by the Distribution Vertical Distance per Feeder multiplied by the Number of Distribution legs multiplied by the density specific percentage of Underground cable.
HG	Underground Distribution Copper Material $	=CopperCostRatio*(IF(GP2=0,0,HF2*VLOOKUP(GP2, DistrCableCost,2,FALSE)+GL2*VLOOKUP(MaxDistrSi ze,DistrCableCost,2))+IF(GS2=0,0,HE2*VLOOKUP(GS 2,DistrCableCost,2,FALSE)+GQ2*VLOOKUP(MaxDistr Size,DistrCableCost,2)))	Underground Distribution Copper material $ equals Total Underground Distribution Vertical Distance multiplied by the Distribution cable cost for the Last Horizontal Distribution Copper Cable Size plus Total Underground Distribution Horizontal Distance multiplied by the cable cost for the # Max Size Horizontal Fiber Distribution Cable Size. The sum of these cable costs are multiplied by 1 minus the copper cable cost discount, if applicable. The formula returns a zero value for the vertical cable cost if the Last Horizontal Distribution Copper Cable Size equals zero and for the Horizontal cable cost if the # Max Horizontal Copper Distribution Cable Size equals zero. Total Distribution Copper Material $ is the sum of Aerial Distribution Copper Material $, Buried Distribution Copper Material $ and Underground Distribution Copper Material $.
HH	Total Distribution Copper Material $	=HA2+HD2+HG2	
HI	Aerial Distribution Fiber Material $	=FiberCostRatio*(IF(GN2=0,0,GZ2*VLOOKUP(GN2,Fi berCableCost,4,FALSE))+IF(GR2=0,0,GY2*VLOOKUP (GR2,FiberCableCost,4,FALSE))+GS2*GY2*VLOOKU P(MaxFiberSize,FiberCableCost,4))	Aerial Distribution Fiber Material $ equals Total Aerial Distribution Vertical Distance multiplied by the cable cost corresponding to Fiber Distribution Leg Cable Size plus Total Aerial Distribution Horizontal Distance multiplied by the cable cost corresponding to the Last Horizontal Fiber Distribution Cable Size plus # Max Size Horizontal Fiber Distribution Cables multiplied Total Aerial Distribution Horizontal Distance multiplied by the cable cost for the maximum size cable. The sum of these cable costs are multiplied by 1 - the fiber cable cost Discount rate, if applicable. The formula returns a zero value for the vertical costs if the Fiber Distribution Leg Cable Size equals zero and for the Horizontal cost if the Last Horizontal Fiber Distribution Cable size equals zero.

Column	Label	Formula	Explanation
HJ	Buried Distribution Fiber Material $	=FiberCostRatio*(IF(GN2=0,0,HC2*VLOOKUP(GN2,FiberCableCost,3,FALSE))+IF(GR2=0,0,HB2*VLOOKUP(GR2,FiberCableCost,3,FALSE))+GS2*HB2*VLOOKUP(MaxFiberSize,FiberCableCost,3))	Buried Distribution Fiber material $ equals Total Buried Distribution Vertical Distance multiplied by the Distribution cable cost for fiber Distribution Leg Cable Size plus Total Buried Distribution Horizontal Distance multiplied by the LastHorizontal Fiber Distribution Cable Size plus #Max Size Horizontal Fiber Distribution Cables multiplied by Total Buried Distribution Horizontal Distance multiplied by the fiber cable cost for the maximum size fiber cable. The sum of these cable costs are multiplied by 1 minus the copper cable cost discount, if applicable. The formula returns a zero value for the vertical cable cost if the Fiber Distribution Leg Cable Size equals zero and for the Horizontal cable cost if the Last Horizontal Distribution Cable Size equals zero.
HK	Underground Distribution Fiber Material $	=FiberCostRatio*(IF(GN2=0,0,HF2*VLOOKUP(GN2,FiberCableCost,2,FALSE))+IF(GR2=0,0,HE2*VLOOKUP(GR2,FiberCableCost,2,FALSE))+GS2*HE2*VLOOKUP(MaxFiberSize,FiberCableCost,2))	Underground Distribution Fiber Material $ equals Total Underground Distribution Vertical Distance multiplied by the Distribution cable cost for the Fiber Distribution Leg Cable Size plus Total Underground Distribution Horizontal Distance multiplied by the cable cost for the Last Horizontal Fiber Distribution Cable Size plus # Max Size Horizontal Fiber Distribution Cables multiplied by Total Underground Distribution Horizontal Distance multiplied by the cable cost for the maximum size fiber cable. The sum of these cable costs are multiplied by 1 minus the copper cable cost discount, if applicable. The formula returns a zero value for the vertical cable cost if the Fiber Distribution Leg Cable Size equals zero and for the Horizontal cable cost if the Last Horizontal Fiber Distribution Cable Size equals zero.
HL	Total Distribution Fiber Material $	=SUM(HI2:HK2)	Total Distribution Fiber Material is the sum of Aerial Distribution Fiber Material $, Buried Distribution Fiber Material $, and Underground Distribution Fiber Material $.
HM	Aerial Distribution Structure $	=IF((GY2+GZ2)<1,0,ROUND((GY2/2+GZ2)/VLOOKUP(U2,SpacingTable,3),1))*AI2*(IF(AH2=1,VLOOKUP(U2,HardRockStructure,7),IF(OR(AH2=2,AF2=1),VLOOKUP(U2,SoftRockStructure,7),VLOOKUP(U2,NormalStructure,7))))	This formula calculates density specific Aerial Distribution structure cost if the sum of Total Aerial Distribution Horizontal Distance and Total Aerial Distribution Vertical Distance do not equal zero. The formula returns a value of zero if the distance equals zero. Structure costs include density specific terrain costs.

Column	Label	Formula	Explanation
HN	Buried Distribution Structure $	=IF((HB2+HC2)<1,0,AI2*(HB2/2+HC2)*IF(AH2=1,VLOOKUP(U2,HardRockStructure,5),IF(OR(AH2=2,AF2=1),VLOOKUP(U2,SoftRockStructure,5),VLOOKUP(U2,NormalStructure,5))))	This formula calculates density specific Buried Distribution structure cost if the sum of Total Buried Distribution Horizontal Distance and Total Buried Distribution Vertical Distance do not equal zero. The formula returns a value of zero if the distances equal zero. Structure costs include density specific terrain costs.
HO	Underground Distribution Structure $	=IF((HE2+HF2)<1,0,AI2*((ROUND((HE2/2+HF2)/VLOOKUP(U2,SpacingTable,2),1)+1)*VLOOKUP(0,ConduitManholeTable,VLOOKUP(AH2,SurfaceConditionTable,2))+AI2*((ROUND((HE2/2)/VLOOKUP(U2,SpacingTable,2),1)+1)*VLOOKUP(IF(FS2>1800,4,0),ConduitManholeTable,VLOOKUP(AH2,SurfaceConditionTable,2))+(HE2+HF2)*(2*Conduitperductfoot)+(HE2/2+HF*IF(AH2=1,VLOOKUP(U2,HardRockStructure,3),IF(OR(AH2=2,AF=1),VLOOKUP(U2,SoftRockStructure,3),VLOOKUP(U2,NormalStructure,3)))))))	This formula calculates density specific Underground Distribution structure cost if Total Underground Distribution Horizontal and Vertical Distances do not equal zero. The formula returns a value of zero if the distances equal zero. Structure costs include density specific terrain costs.
HP	Total Distribution Structure $	=SUM(HM2:HO2)	Total Distribution Structure $ is the sum of Aerial, Buried and Underground Structure $.
HQ	Drop Cost	=IF(VLOOKUP(U2,DensityHhTable,3)>3,GX2*(K2*VLOOKUP(U2,DensityHhTable,2)*DropCostPerFoot+(K2*VLOOKUP(U2,DensityHhTable,4))/VLOOKUP(U2,DensityHhTable,3)*((3.06/VLOOKUP(U2,DensityHhTable,3))-0.002)),IF(GX2>500,500*K2*DropCostPerFoot,GX2*K2*DropCostPerFoot))	Drop costs are determined separately for single family dwellings and multi-unit dwellings with more than 3 house holds per dwelling. Multi-unit dwellings of 3 or fewer house holds per unit are treated as single family units. The drop cost is based on the Length To Center of Lot (GX2) which is limited to a maximum of 500 feet. The cost for multi-family dwellings is a function of 25 and 50 pair cables.
HR	Network Interface Device	=IF(VLOOKUP(U2,DensityHhTable,3)>3,K2*VLOOKUP(U2,DensityHhTable,2)*NidCost,K2*NidCost)	Network Interface Device cost (NID) is developed for multi-family units with more than 3 households per dwelling unit and single family dwelling units. The cost of NID is calculated by multiplying the appropriate number of units (house holds) by NidCost.
HS	Single Family Pedestal, Terminal and Splicing	=IF(OR(GW2=0,K2=0),0,IF(V2/GW2<50,GW2*(PedestalCost+VLOOKUP(U2,DensityHhTable,2)*(VLOOKUP(U2,DistributionPlantMixTable,4)*VLOOKUP(K2*ResLinesMultiplier/GW2,DropTerminalCostTable,3))+VLOOKUP(U2,DensityHhTable,2)*(1-VLOOKUP(U2,DistributionPlantMixTable,4))*VLOOK	Single Family Pedestal, Terminal and splicing costs are developed separately for CBGs with fewer than 50 lines per pedestal and those with 50 or more. In CBGs with fewer than 50 lines per pedestal the Pedestal cost is added to Drop terminal cost which is split between aerial and buried since the cost vary according to distribution plant mix (underground, buried and aerial). In CBGs with 50 0r more

Column	Label	Formula	Explanation
		UP(K2*ResLinesMultiplier/GW2,DropTerminalCostTable,2)),GW2*VLOOKUP(U2,DensityHhTable,2)*VLOOKUP(K2*ResLinesMultiplier/GW2,DropTerminalCostTable,4)))	lines per pedestal the cost is determined by multiplying the number of pedestals for single family dwellings by the appropriate drop terminal cost.
HT	Number of Maximum Multi-Family & Business Terminals	=IF(V2=0,0,IF(V2/GW2>900,GW2*TRUNC((V2/GW2)/900),0))	The maximum number of multi-family and business terminals is calculated by multiplying the number of pedestals in the CGG by the number of maximum sized terminals. The largest terminal serves 900 lines.
HU	Multi-Family & Business Pedestal, Terminal and Splicing	=IF(V2=0,0,IF(K2=0,GW2*VLOOKUP((V2/GW2)-900*HT2,DropTerminalCostTable,4),K2*VLOOKUP(U2,DensityHhTable,4)/VLOOKUP(U2,DensityHhTable,3)*VLOOKUP((V2-(900*HT2)-(K2*ResLinesMultiplier*VLOOKUP(U2,DensityHhTable,2)))/(K2*VLOOKUP(U2,DensityHhTable,4))/VLOOKUP(U2,DensityHhTable,3),DropTerminalCostTable,4))+HT2*VLOOKUP(900,DropTerminalCostTable,4))	Multi-family and business pedestal, terminal and splicing cost are determined for the correct size drop terminal.
HV	Aerial Copper Cable Investment $	=IF(HH2+HL2=0,0,CN2+DF2+FU2+HA2+(HQ2+HR2+HS2+HU2)*HA2/(HH2+HL2))	Aerial Copper Cable Investment $ = Main Feeder Segment B Installed Aerial Copper $ + Subfeeder Aerial Cable Segment A Copper $ + Aerial Copper Feeder Extension Part 2 Material $ + (Drop Cost+Network Interface Device+Pedestal, Terminal and and Splicing)*Aerial Distribution Copper Material $/(Total Distribution Copper Material $ + Total Distribution Fiber Material $)
HW	Buried Copper Cable Investment $	=IF(HH2+HL2=0,0,CL2+CU2*BF2+DD2+FX2+HD2+IF(AW2="Cable",DM2+GF2+HN2+EY2,0)+(HQ2+HR2+HS2+HU2)*HD2/(HH2+HL2))	Buried Copper Cable Investment $ = Main Feeder Segment B Installed Buried Copper $ + Main Feeder Buried Segment B Total Structure $ * Copper Cable Structure % + Subfeeder Buried Cable segment A Copper $ + Buried Copper Feeder Extension Part 2 Material $ + Buried Distribution Copper Material $ + (if Segment Type 1=Cable then Total Subfeeder Segment A Buried Structure $ + Buried Distribution Structure $ + Feeder Distribution Interface Cost, then zero if not Cable) + ((Drop Cost+Network Interface Device+Pedestal, Terminal and Splicing)*Buried Distribution Copper Material $/(Total Distribution Copper Material $))
HX	Underground Copper Cable Investment $	=IF(HH2+HL2=0,0,CJ2+DB2+GA2+HG2+(HQ2+HR2+HS2+HU2)*HG2/(HH2+HL2))	Underground Copper Cable Investment $ = Main Feeder Segment B Installed Underground Copper $ + Subfeeder Underground Cable

Column	Label	Formula	Explanation
			Segment A Copper $ + Underground Copper Feeder Extension Part 2 Material $+ Underground Distribution Copper Material $ + ((Drop Cost+Network Interface Device+Pedestal, Terminal and Splicing)*Underground Distribution Copper Material $/(Total Distribution Copper Material $ + Total Distribution Fiber Material $))
HY	Pole Investment $ for copper	=CV2*BF2+HM2+IF(AW2="Cable",DN2+FL2+GE2,0)	Pole Investment $ for copper = Main Feeder Aerial segment B Total Structure $ * Copper Cable Structure % + Aerial Distribution Structure $ + (if Segment Type 1=Cable, Total Subfeedeer Segment A Aerial Structure $ + Aerial Feeder Extension Part 1 Structure $ + Aerial Feeder Extension Part 2 Structure $, then zero if not cable)
HZ	Conduit Investment $ for copper	=CT2*BF2+HO2+IF(AW2="Cable",DL2+FN2+GG2,0)	Conduit Investment $ for copper = Main Feeder Underground Segment B Total Structure $ * Copper Cable Structure % + Underground Distribution Structure $ + (if Segment Type 1=Cable, Total Subfeedeer Segment A Underground Structure $+ Underground Feeder Extension Part 1 Structure $ + Underground Feeder Extension Part 2 Structure $, then zero if not cable)
IA	Copper related plant total	=SUM(HV2:HZ2)	Copper related plant total = Aerial Copper Cable Investment $ + Buried Copper Cable Investment $ + Underground Copper Cable Investment $ + Pole Investment $ for copper + Conduit Investment $ for copper
IB	Aerial Fiber Cable Investment $	=IF(HH2+HL2=0,0,DW2+EL2+FF2+FV2+HI2+(HQ2+HR2+HS2+HU2)*HI2/(HH2+HL2))	Aerial Fiber Cable Investment $ = Aerial Fiber Main Feeder Material $ + Aerial Fiber Subfeeder Material $ + Aerial Fiber Feeder Extension Part 1 Material $+ Aerial Fiber Feeder Extension Part 2 Material $ Aerial Distribution Fiber Material $+ ((Drop Cost+Network Interface Device+Pedestal, Terminal and Splicing)*Aerial Distribution Fiber Material $/(Total Distribution Copper Material $ + Total Distribution Fiber Material $))
IC	Buried Fiber Cable Investment $	=IF(HH2+HL2=0,0,DU2+CU2*BG2+EJ2+FH2+FM2+FY2+HJ2+IF(NOT(AW2="Cable"),DM2+GF2+HN2+EY2,0)+(HQ2+HR2+HS2+HU2)*HJ2/(HH2+HL2))	Buried Fiber Cable Investment $ = Buried Fiber Main Feeder Material $ + Main Feeder Buried Segment B Total Structure $ * Fiber Structure % + Buried Fiber Subfeeder Material $ + Buried Fiber Feeder Extension Part 1 Material $ + Buried Feeder Extension Part 1 Structure $ + Buried Fiber Feeder Extension Part 2 Material $ + Buried Distribution Fiber Material $ + (if Segment Type 1 does

Column	Label	Formula	Explanation
			not=Cable then Total Subfeeder Segment A Buried Structure $ + Buried Feeder Extension Part 2 Structure $
			+ Buried Distribution Structure $ + Feeder Distribution Interface Cost, then zero if Cable) + ((Drop Cost+Network Interface Device+Pedestal, Terminal and Splicing)*Buried Distribution Fiber Material $/(Total Distribution Copper Material $ + Total Distribution Fiber Material $)
ID	Underground Fiber Cable Investment $	=IF(HH2+HL2=0,0,DS2+EH2+FJ2+GB2+HK2+(HQ2+HR2+HS2+HU2)*HK2/(HH2+HL2))	Underground Fiber Cable Investment $ = Underground Fiber Main Feeder Material $ + Underground Fiber Subfeeder Material $ + Underground Fiber Feeder Extension Part 1 Material $+ Underground Fiber Feeder Extension Part 2 Material $ + Underground Distribution Fiber Material $ + ((Drop Cost+Network Interface Device+Pedestal, Terminal and Splicing)*Underground Distribution Fiber Material $/(Total Distribution Copper Material $ + Total Distribution Fiber Material $))
IE	Pole Line Investment $ for fiber	=EA2+IF(AW2="Cable",0,DN2+FL2+GE2)	Pole Line Investment $ for Fiber = Aerial Fiber Main Feeder Structure $ + ((Total Subfeeder Segment A Aerial Structure $ + Aerial Feeder Extension Part 1 Structure $ + Aerial Feeder Extension Part 2 Structure $) unless Segment Type 1 = "Cable". If Segment Type 1 is Cable then these component investments equal zero.)
IF	Conduit Investment $ for fiber	=DY2+IF(AW2="Cable",0,DL2+FN2+GG2)	Conduit Investment $ for Fiber = Underground Fiber Main Feeder Structure $ + ((Total Subfeeder Segment A Underground Structure $ + Underground Feeder Extension Part 1 Structure $ + Underground Feeder Extension Part 2 Structure $) unless Segment Type 1 ="Cable". If Segment Type 1 is Cable then these component investments equal zero.)
IG	Total Fiber Related Plant	=SUM(IBY2:IF2)	Fiber Related Plant Total = Aerial Fiber Cable Investment $ + Buried Fiber Cable Investment $ + Underground Fiber Cable Investment $ + Pole Line Investment $ for Fiber + Conduit Investment $ for Fiber
IH	Intentionally Blank		

Column	Label	Formula	Explanation
II	Clli	=A2	Used for labeling switch cost by CLLI.
IJ	Total Lines Equipped in Switch	=SUM(CG:CG)	This equals switched lines equipped.
IK	Fixed Cost/Ln	=IF(IJ2=0,0,((VLOOKUP(C2,COSwitchCost,2,FALSE)*TrfSen)/IJ2*SwitchFillFactor)*(1+(VLOOKUP(C2,COSwitchCost,4,FALSE)+VLOOKUP(C2,COSwitchCost,5,FALSE))))	This determines the fixed cost per line based on the switch cost associated with the size of the company (large, medium or small). The fixed costs include Power and Common equipment plus Telco Install and Engineering.
IL	Variable Cost per Line	=(VLOOKUP(C2,COSwitchCost,3,FALSE)/SwitchFillFactor*TrfSen)*(1+VLOOKUP(C2,COSwitchCost,4,FALSE)+VLOOKUP(C2,COSwitchCost,5,FALSE))	The variable cost per line are based per line cost according to company size (large, medium or small)
IM	Total Line Cost	=SUM(IK2:IL2)	This is the sum of fixed cost per line and variable cost per line.

DOCKET FILE COPY ORIGINAL

Visual Basic Module Sheet Descriptions for the BCPM

ModControl

The module contains an assortment of general use functions. The main purpose of these functions is to handle moving between workbooks, general housekeeping, and formatting of worksheet tabs. This sheet also contains the Auto_Open() and Auto_Close() routines.

ModFileHandling

The module contains just 3 functions. BuildFileIndexes() generates an "ini" type index file for the input and output files of the BCPM model. ExtractName() and ExtractPath() return the name and the path of a given filename.

ModIniFileCode

This module contains a collection of generic functions for dealing with the "ini" files. This file format is used extensively throughout the model.

ModInputTables

This module contains a collection of functions that handle inputting data into the BCPM model.

ModLicenseAgreement

This module contains one function to handle the response to the License Agreement question.

ModMenus

This module contains functions that control the application defined menus and the updating of the displays when sheet tabs are changed.

ModModel

This module contains functions used to create and modify Views. This code is generally associated with the "dlgNewModel" worksheet.

ModOptions

This module contains two functions that control the "Options" worksheet.

ModProcess

This module contains a collection of functions that handles the actual processing of the input data through the BCPM logic set. The functions to parse the input records, move the ASCII data into the "Main Logic" tab, and write out the results reside here.

ModReports

The module contains a collection of functions that control populating the "Reports" worksheet and moving the report data into the Reports workbook.

ModRollup

The module contains a collection of functions that prepares the model output data for the reports. The ApplyAggregateSupport() function generates aggregate support for each CBG. All cumulation of data is done by the functions in this module.

ModIniFileStuff

This module contains a collection of generic functions for dealing with the "ini" files. This file format is used extensively throughout the model.

ModMain

This module contains a collection of functions that control processing within this workbook. The functions generally handle menu creation, responses to menu choices, and writing the output data.

```
Option Explicit: Option Base 1: Option Compare Text

Private Const MENU_TITLE = "&BCPM"

'================================================================================
============
Public Sub CreateMenus()
'   Adds the BCP Model menus
Dim index           As Integer, _
    MenuNames(2)    As Integer
Dim BCPMenu         As Menu
Dim tempMenuItem    As Object

'   --Erase any existing menus
    RemoveMenus

'   --only impact worksheet, module menus
    MenuNames(1) = xlWorksheet
    MenuNames(2) = xlModule

'   --add the menu
    For index = 1 To 2
        Application.MenuBars(MenuNames(index)).Menus.Add _
            before:="&Help", Caption:=MENU_TITLE
        Set BCPMenu = Application.MenuBars(MenuNames(index)).Menus(MENU_TITLE)
        With BCPMenu.MenuItems
            .Add Caption:="Process", _
                 OnAction:="SwitchToProcessTab", _
                 StatusBar:="Process selected Census Block Group data"
            .Add Caption:="Review", _
                 OnAction:="SwitchToReviewTab", _
                 StatusBar:="Review selected Wire Center data"
            .Add Caption:="Reports", _
                 OnAction:="SwitchToReportsTab", _
                 StatusBar:="Review/Print reports"
            .Add Caption:="Instructions", _
                OnAction:="SwitchToInstructionTab", _
                StatusBar:="Show brief instructions for major functions"
            .Add Caption:="Input Tables", _
                 OnAction:="SwitchToInputTablesTab", _
                 StatusBar:="Edit manual inputs, create an alternate logic set"
            .Add Caption:="Options", _
                 OnAction:="SwitchToOptionsTab", _
                 StatusBar:="Set options for printine, aggregate support calculation
'
            .Add Caption:="-"
            .Add Caption:="Print", _
                 OnAction:="PrintSelectedWorksheets", _
                 StatusBar:="Print selected worksheets from active workbook"
            .Add Caption:="-"
            .Add Caption:="Capital Cost Factors", _
                 OnAction:="mnuCapCost", _
                 StatusBar:="Edit/Create Capital Cost Views"
            .Add Caption:="-"
            .Add Caption:="Close BCPM", _
                 OnAction:="mnuExit", _
                 StatusBar:="Close model, return to Excel"
        End With
    Next index

End Sub
```

```
Public Sub SwitchToInputTablesTab()

  Application.DisplayFormulaBar = True
  Windows("Control").Activate
  InstructionsSheet_GotFocus
  Sheets("Input Tables").Activate

End Sub

Public Sub SwitchToInstructionTab()

  Windows("Control").Activate
  InstructionsSheet_GotFocus
  Sheets("Instructions").Activate

End Sub

Public Sub SwitchToReviewTab()

  Windows("Control").Activate
  ReviewSheet_GotFocus
  Sheets("Review").Activate

End Sub

Public Sub SwitchToOptionsTab()

  Windows("Control").Activate
  OptionsSheet_GotFocus
  Sheets("Options").Activate

End Sub

Public Sub SwitchToReportsTab()

  Windows("Control").Activate
  ReportsSheet_GotFocus
  Sheets("Reports").Activate

End Sub

Public Sub SwitchToProcessTab()

  Windows("Control").Activate
  ProcessSheet_GotFocus
  Sheets("Process").Activate

End Sub


Public Sub mnuCapCost()
Dim dlg                 As DialogSheet
Dim lst                 As Listbox
Dim FileSpec            As String, _
    FactorPath          As String

  Set dlg = ThisWorkbook.DialogSheets("dlgCapCostWorkbooks")
  Set lst = dlg.ListBoxes("lstWorkbooks")

  lst.RemoveAllItems
```

```
  FactorPath = ThisWorkbook.Sheets("RangeNames").Range("Factor_Path")
  FileSpec = FactorPath & "*.xls"

  If GetFileNames(FileSpec) Then
    lst.List = p_FileNames
    If dlg.Show Then
      If lst.ListIndex Then
        Workbooks.Open FileName:=FactorPath & lst.List(lst.ListIndex)
        ActiveWorkbook.RunAutoMacros xlAutoOpen
      End If
    End If
  End If

  Set dlg = Nothing
  Set lst = Nothing

End Sub

Public Sub PrintSelectedWorksheets()
Dim dlg                 As DialogSheet
Dim lstSheets           As Listbox
Dim obj                 As Object
Dim index               As Long
Dim FromSheet           As Worksheet


  Set dlg = ThisWorkbook.DialogSheets("dlgPrintWorksheets")
  Set lstSheets = dlg.ListBoxes("lstWorksheets")

  lstSheets.RemoveAllItems

  If ActiveWorkbook.Name = ThisWorkbook.Name Then
    lstSheets.AddItem "Status"
  Else
    For Each obj In ActiveWorkbook.Worksheets
      lstSheets.AddItem obj.Name
    Next obj
  End If

  If dlg.Show Then
    If lstSheets.ListCount Then
      Set FromSheet = ActiveWorkbook.ActiveSheet
      For index = 1 To lstSheets.ListCount
        If lstSheets.Selected(index) Then
          With ActiveWorkbook.ActiveSheet.PageSetup
            If Application.OperatingSystem <> "Windows (32-bit) 4.00" Then
              .LeftFooter = "&F.xls"
            Else
              .LeftFooter = "&F"
            End If
            .CenterFooter = "Page &P"
            .RightFooter = "&D &T"
          End With
          ActiveWorkbook.Worksheets(lstSheets.List(index)).PrintOut
        End If
      Next index
      FromSheet.Activate
    End If
  End If
```

```
  Set dlg = Nothing
  Set lstSheets = Nothing
  Set FromSheet = Nothing

End Sub

Public Sub mnuExit()

  CloseAllOpenWorkbooks
  CloseControlWorkbook

End Sub

'=============================================================================
============
Public Sub RemoveMenus()
Dim index           As Integer
Dim MenuNames(3)    As Integer

    MenuNames(1) = xlWorksheet
    MenuNames(2) = xlModule
    MenuNames(3) = xlNoDocuments

    On Error Resume Next
    For index = 1 To 3
        Application.MenuBars(MenuNames(index)).Menus(MENU_TITLE).Delete
    Next index

End Sub
```

```
Option Explicit: Option Base 1: Option Compare Text

'32-bit API declarations
Private Declare Function GetPrivateProfileInt& Lib "kernel32" Alias "GetPrivateProfi
leIntA" (ByVal AppName As String, ByVal KeyName As String, ByVal Default As Long, By
Val FileName As String)
Private Declare Function GetPrivateProfileSection& Lib "kernel32" Alias "GetPrivateP
rofileSectionA" (ByVal AppName As String, ByVal ReturnedString As String, ByVal Stri
ngSize As Long, ByVal FileName As String)

Public m_IndexFile       As String


'=============================================================================
============
' BuildFileIndexes: This function build an index file for the various files.  The ou
tput file is
'                   in "ini" file format.  The index is a set of starting record num
bers for each
'                   Wire Center.
Public Function BuildFileIndexes(FileName As String, Optional columns As Variant) As
 String
Dim InputStream         As Integer, _
    position            As Integer
Dim buffer              As String, _
    IndexFile           As String
Dim HoldParent          As String, _
    HoldCompany         As String, _
    HoldWireCenter      As String
Dim ParentCol           As Long, _
    CompanyCol          As Long, _
    WireCenterCol       As Long, _
    RecordLocation      As Long
Dim Record              As Variant
Dim PassColumns         As Boolean

  PassColumns = Not IsMissing(columns)
  ShowStatusMessage "building index files"

' --get the column numbers
  IndexFile = ThisWorkbook.Path & "\models.inf"
  ParentCol = GetIniLong("Parent", "Input Columns", IndexFile)
  CompanyCol = GetIniLong("Company")
  WireCenterCol = GetIniLong("CLLI")

' --open the input file
  InputStream = FreeFile()
  Open FileName For Input Shared As #InputStream

' --build the output file name, assumes DOS file naming convention
  position = InStr(FileName, ".")
  IndexFile = Left$(FileName, position) & "idx"

' --initialize variables
  HoldWireCenter = ""
  HoldParent = ""
  HoldCompany = ""
  RecordLocation = 0

' --delete all sections
  DeleteSection "Wire Centers", IndexFile
```